

## **РЕШЕНИЕ МАТЕМАТИЧЕСКИХ ЗАДАЧ НА ЯЗЫКЕ VISUAL C# С ИСПОЛЬЗОВАНИЕМ ПАКЕТА MATLAB**

### **Аннотация**

*В работе рассмотрены особенности и этапы интеграции программного обеспечения, написанного на языке программирования Visual C#, с математическим пакетом MATLAB на примере решения задачи линейного программирования. Приведены листинги программ в MATLAB и Visual C#. Сделаны выводы о преимуществах и недостатках изложенного решения.*

*Ключевые слова: математическое программирование, язык программирования, интеграция, Visual C#, MATLAB, компилятор.*

### **Abstract**

*The article deals with the features and integration stages application in Visual C# and the mathematical package MATLAB. Solution overview organized on the example solving a linear programming problem. Listings of programs in MATLAB and Visual C# given. Conclusions about the advantages and disadvantages of the above solutions done.*

*Keywords: mathematical programming, programming language, integration, Visual C#, MATLAB, compiler.*

### **Введение**

Современные языки разработки программного обеспечения под платформу Windows, например, Visual C#, позволяют в короткие сроки создавать стильные и функциональные приложения. Однако эти средства не содержат в себе встроенных математических библиотек, которые необходимы при разработке приложений для решения научно-практических задач, а сторонние математические библиотеки, такие как Extreme Optimization, IMSL Numerical Library и др., имеют высокую лицензионную стоимость. С другой стороны, инженерные пакеты с большим количеством встроенных математических библиотек и функций, например, MathCAD и MATLAB, не включают средства для создания удобного пользовательского интерфейса и не позволяют разрабатывать функциональные приложения. В связи с этим, хотя подход интеграции языка программирования с инженерным пакетом и является нетривиальной задачей, он весьма интересен при решении широкого круга прикладных задач.

Рассмотрим технологию интеграции языка программирования Visual C# с MATLAB R2012b на примере решения задачи линейного программирования. Работа выполнена в рамках разработки программного обеспечения, предназначенного для решения задачи оптимального распределения природного газа в группе доменных печей с учетом индивидуальных технологических ограничений на каждую печь и параметров работы доменного цеха, и состоит из двух этапов. На первом этапе осуществляется создание математической библиотеки, которая выполняет решение задачи, а на втором её подключение к рабочей программе.

### **Создание математической библиотеки**

Отличительной особенностью MATLAB от других математических пакетов является наличие средств, которые позволяют использовать функционал, созданный в MATLAB, при разработке автономных приложений. Такая возможность реализуется с помощью продукта MATLAB Compiler. Данный продукт позволяет собирать модули в виде [1]:

- независимое приложение;
- C/C++ библиотеки (DLL в Windows, разделяемые библиотеки в Linux);

– программные компоненты, такие как классы Java, сборки .NET или надстройки Microsoft Excel для использования в других приложениях.

Приложения и библиотеки, созданные с помощью MATLAB Compiler, используют свободно распространяемое исполняемое ядро, называемое MATLAB Compiler Runtime (MCR). Это позволяет запускать приложения, которые используют MATLAB-функции, без необходимости установки дорогостоящей копии MATLAB.

Одним из инструментов для создания математических библиотек или внешних компонентов является Deployment Tool, представляющий собой графический интерфейс. Вызов данного инструмента осуществляется командой `deploytool`. На рисунке 1 представлено стартовое окно Deployment Tool, на котором предлагается выбрать тип и создать новый проект.

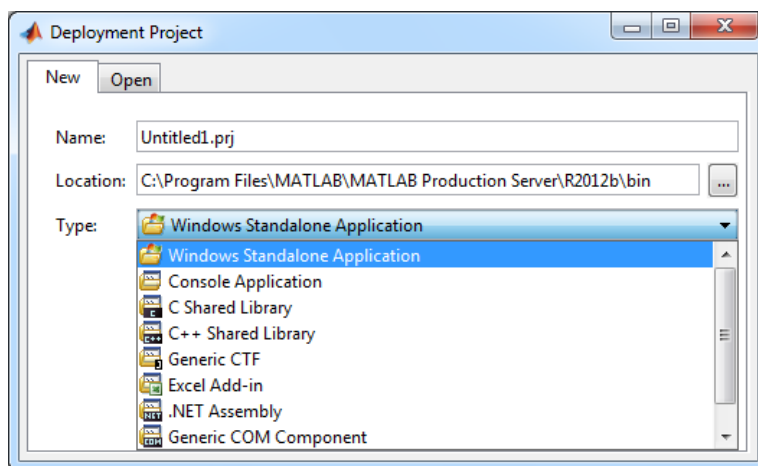


Рис. 1. Стартовое окно инструмента Deployment Tool в MATLAB

Название проекта определяет пространство имен, используемое в дальнейшем в Visual C#. Проект состоит из классов, которые в свою очередь включают М-файлы. Каждый М-файл включает одну функцию, которая вызывается как метод соответствующего класса в Visual C#.

Рассмотрим решение задачи линейного программирования в пакете MATLAB. Решение осуществляется с помощью функции `linprog` [3]. Функция `linprog` решает задачу линейного программирования в форме:

$$\begin{aligned} f^T \cdot x &\rightarrow \inf, \\ A \cdot x &\leq b, \\ Aeq \cdot x &\rightarrow beq, \\ lb &\leq x \leq ub. \end{aligned} \tag{1}$$

Основными входными данными функции `linprog` являются: вектор коэффициентов целевой функции `f`, матрица ограничений-неравенств `A`, вектор правых частей ограничений-неравенств `b`, матрица ограничений-равенств `Aeq`, вектор правых частей ограничений-равенств `beq`, вектор `lb`, ограничивающий план `x` снизу, вектор `ub`, ограничивающий план `x` сверху. На выходе функция `linprog` даёт оптимальный план `x` задачи (1) и экстремальное значение целевой функции `fval`.

Листинг функции `MOptimaGaz` решения задачи линейного программирования:

```
function [x, fval] = MOptimaGaz(f, A, b, lb, ub)
    Aeq = [];
    beq = [];
    options = optimset('LargeScale', 'off', 'Simplex', 'on');
    [x, fval] = linprog(f, A, b, Aeq, beq, lb, ub, [], options);
end
```

Стоит заметить, что при решении практической задачи не использовались ограничения-равенства, поэтому матрица `Aeq` и вектор `beq` не передаются как входные параметры, а принимают пустые значения.

Таким образом, в MATLAB был создан проект *MOptimaGaz* с классом *MClassOptimaGaz* и функцией *MOptimaGaz*. Для использования написанных функций проект компилируется в динамически подключаемую библиотеку DLL.

### Подключение библиотеки и ее использование

Для использования внешней библиотеки среда разработки Visual Studio 2012 и язык программирования Visual C# предусматривает механизм ссылок на сборку. Для подключения написанных в MATLAB функций необходимо добавить ссылку на созданную библиотеку, а также на библиотеку *MWArray*, находящуюся в папке установленной программы по адресу “C:\Program Files\MATLAB\MATLAB Production Server\R2012b\toolbox\dotnetbuilder\bin\win32\v2.0\MWArray.dll”. На рисунке 2 представлено окно добавления новой ссылки на сборку.

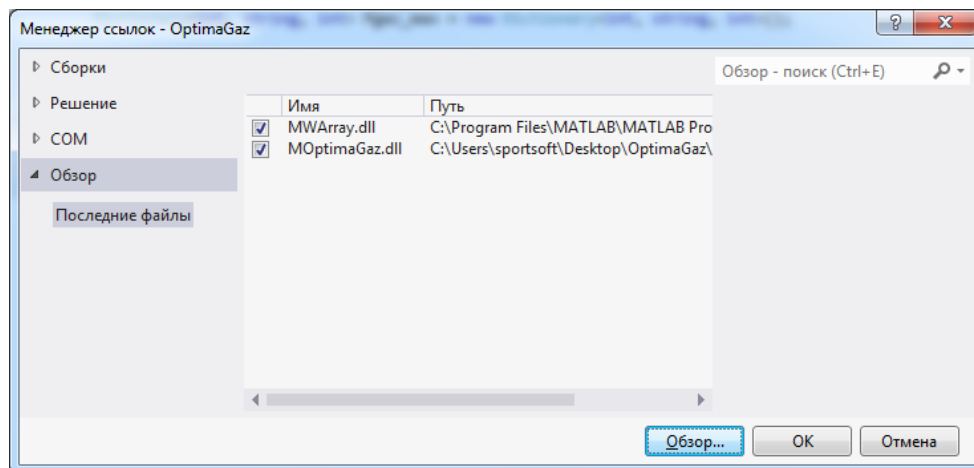


Рис. 2. Окно добавления ссылки на сборку в Visual Studio

Библиотека *MWArray* предназначена для обмена данными между средой программирования MATLAB и платформой .NET. Она обеспечивает преобразование типов данных, отслеживание ошибок выполнения кода, экспорт графики в приложения ASP.NET и др. [2].

Отметим обязательность соблюдения разрядности MATLAB и создаваемого программного обеспечения. Также замечена нестабильность работы при работе с 64-разрядной архитектурой, поэтому при разработке использовалась 32-разрядная версия MATLAB.

Для использования библиотеки необходимо добавить описание пространства имен:

```
using MathWorks.MATLAB.NET.Utility;
using MathWorks.MATLAB.NET.Arrays;
using MOptimaGaz;
```

Ниже приведен листинг программы с комментариями, вызывающий внешнюю функцию MATLAB и выводящий результаты решения задачи.

```
MClassOptimaGaz MObject = new MClassOptimaGaz();

MWNumericArray descriptor = null; // Массив возвращаемого параметра

int x = 8; // Количество неизвестных задачи оптимизации
double[] lb = new double[x];
double[] ub = new double[x];
double[] f = new double[x];

double[,] Aeq = new double[3, x]; // Всего 3 ограничения-неравенства
double[] beq = new double[3]; // Всего 3 ограничения-неравенства

// Присваивание переменных lb, ub, Aeq, beq и f реальных значений

MWArray[] result = MObject.MOptimaGaz(2, (MWNumericArray)f, (MWNumericArray)Aeq,
(MWNumericArray)beq, (MWNumericArray)lb, (MWNumericArray)ub);
```

*// Первый параметр - это количество возвращаемых аргументов*

```
for (int j = 0; j < result.Length; j++)
{
    MessageBox.Show("Вывод массива " + j.ToString());

    descriptor = (MWNumericArray)result[j];
    double[,] d_descriptor = (double[,])descriptor.ToArray(MWArrayComponent.Real);
    // Преобразование массива MWNumericArray к массиву типа double

    for (int i = 0; i < d_descriptor.Length; i++)
    {
        MessageBox.Show(d_descriptor[i, 0].ToString());
    }
}
```

Из программы опущен блок, присваивающий переменным lb, ub, Aeq, beq и f значения. Если решение не найдено, то результат содержит пустой массив.

### **Заключение**

Подход интеграции языка программирования Visual C# и математического пакета MATLAB позволяет разрабатывать стильные и удобные пользовательские интерфейсы в сочетании с мощным инструментом для решения математических задач. Преимуществами такого подхода в сравнении с другими инженерными пакетами являются отсутствие необходимости установки копии MATLAB для запуска создаваемых программ, возможность использования создаваемых библиотек в разных проектах. Замеченным недостатком является время задержки (около одной секунды) при вызове внешних функций.

### **Список использованных источников**

1. MATLAB Compiler [Электронный ресурс] // Официальный дистрибьютор MathWorks на территории России и СНГ: [web-сайт]. – Режим доступа: <http://matlab.ru/products/matlab-compiler> (дата обращения 01.03.2016).
2. MWArrayAPI Help [Электронный ресурс] // The MathWorks, Inc.: [web-сайт]. – Режим доступа: <http://www.mathworks.com/help/dotnetbuilder/MWArrayAPI/html/57290a83-9e6e-4179-a1b8-9fb0344544ce.htm> (дата обращения 01.03.2016).
3. Кетков Ю.Л. MATLAB 7: программирование, численные методы. – Спб.: БХВ-Петербург, 2005. – 752 с.